

- Notes :
- don't use any electronic device.
 - move away anything related or unrelated to the course away from your seat.
 - do not use pencil.
 - the answer of Q1,Q4,Q5 must be written in a first page
 - other questions , each Question must be in a single page .

1-List the five control and status signals available in 8086?

ALE : active high signal

DT/R' : during read cycle this pin is zero, while during write bus cycle this pin is high "1"

wr' : write control signal , which is active low

rd' : read control signal which is active low.

DEN': data enable control signal which is active low signal.

2-How many bus cycle are required for the following instruction CALL [AX], and Draw/ explain the timing diagram of the instruction, in case of 8086 used and Ax=FA4C?

- one read bus cycle to get the address of the sub-procedure from memory location FA4C

" by default near sub-procedure unless a FAR directive has been use"

- one write bus cycle , to push the IP to the stack.

an onther accepted solution :

[AX] pointing to a 32bit value " IP &CS" so we need to read bus cycle.

and two write bus cycle to push IP and Cs to the stack.

3- A 8086's microprocessor interfaced to an digital temperature sensor via port number 55h,
Write an assembly program to do the following :

- Reads the temperature (sample) from the sensor once the NMI has been activated by the sensor.
- Display the temperature on screen, the temperature must be displayed at the center of the screen.
- Save the temperature in temporary array "array_sample", after you get 900 sample ,get the average of the last 900 sample , check the average temperature if its above 40c should display the following message " warning temp is high " and store it in different array "avg_array".

xxx segment

str db "" warning temp is high\$"

array_sample db 900d dup 00h

avg_array dw xxx dup 0000h

xxx ends

YYY SEGMENT

assume cs:yyy; ds:xxx

Start :

xor ax ,ax

MOV DI , 08H

MOV ES:[DI], OFFSET XIP

MOV ES:[DI+2] , SEGMENT YCS

MOV BX , OFFSET ARRAY_SAMPLE

mov si , offset avg_array

```
DEC BX
MOV BP ,00 ; counter for number of samples
```

```
Mov si , offset str
```

```
Here : jmp here
```

```
RET
```

```
YYY ENDS
```

```
END START
```

```
=====
; ISR code
```

```
YCS SEGEMENT
```

```
ASSUME CS:YCS , DS:XIP
```

```
XIP:
```

```
IN AL , 55H ; TEMP IS NOT MORE THAN +127 OR LES -128 i.e 8 bit value
```

```
; write clear screen code here
```

```
; set curser position at the center
```

```
MOV AH 02H
```

```
MOV DL ,AL
```

```
INT 21H
```

```
INC BX ;add index one
```

```
MOV [BX],AL ; STORE THE TEMP IN TEMP_ARRAY
```

```
INC BP
```

```
CMP BP , 900d
```

```
JNE x
```

```
; set curser one line below the center
```

```
MOV ax ,00h
```

```
mov cx ,900d
```

```
y: add AL ,[bx]
```

```
JNC NEXT
```

```
ADD AH ,1
```

```
NEXT : DEC BX
```

```
LOOP Y
```

```
CWD
```

```
IDIV 900d
```

```
mov [si] , ax
```

```
inc si
```

```
cmp ax ,40
```

```
JB x
```

```
mov dx ,offset str
```

```
mov ah, 09h
```

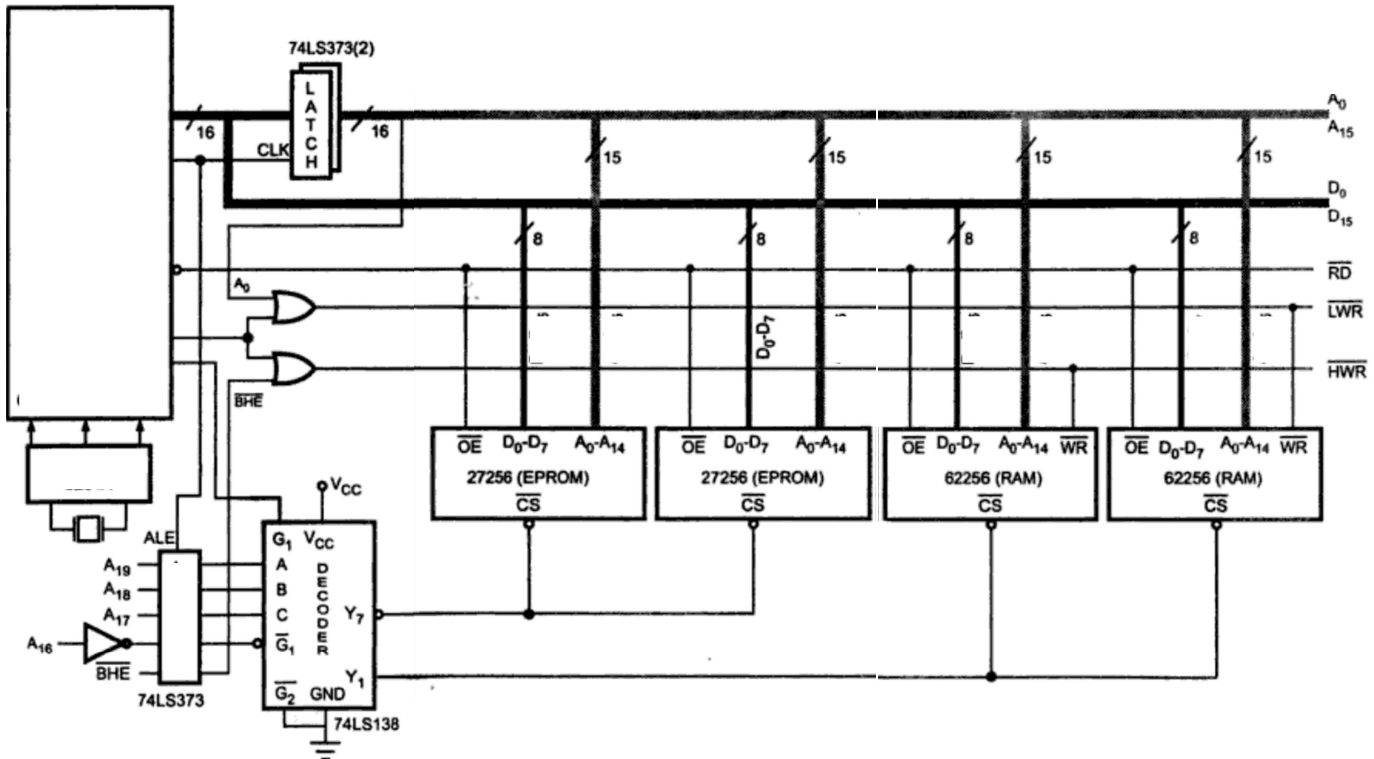
```
INT 21h
```

```
x: iert
```

```
Ycs ends
```

```
end xip
```

1- What is the microprocessor, memory size, memory chip size, address range for each chip, and any other useful information?



MICROPROCESSOR IS 8086

MEMORY SIZE IS 64KB ROM AND 64KB RAM

EACH MEMORY CHIP SIZE IS 32KB

ADDRESS RANGE FOR BOTH RAM CHIPS 30000H- 3FFFFH

ADDRESS RANGE FOR BOTH RAM CHIPS F0000H- FFFFFH

5- Which interrupts are generally used for critical events? **NMI**

6- Write complete ALP to find factorial of number for 8086?

.DATA

STR DB "PLEASE CHOSE SMALLER NUMBERS"

NUM DB XX ;XXX IS THE NUMER YOU WANT TO FIND ITS FACTORIAL

; ASSUME THE NUMBER IS SMALL

;"OTHER OPTION" YOU CAN ENTER THE NUMBER THROUGH INT21

;FROM KEYBOARD

.CODE

.STRTUP

XOR CX,CX

XOR AX,AX

MOV AL,NUM

DEC NUM

MOV CL,NUM

X: MUL NUM

JO Y

LOOP X

Y: MOV AH,09H

MOV DX, OFFSET STR

INT 21H

.EXIT

RET

7- Write complete ALP to reverse a given string for 8086?

```
.DATA
STR DB "PLEASE CHOSE SMALLER NUMBERS$"
      ; ASSUME WE HAVE THIS STRING WANT REVERS IT

.CODE
.STRATUP
MOV BX,OFFSET STR
MOV AH,09H
MOV DX, OFFSET STR
INT 21H
      ; COMPUTE THE NUMBER OF ELEMENT
MOV SI, 00H
Y: MOV AL, [BX+SI]
   CMP AL, '$'
   JE  X
   INC SI
   JMP Y
X: MOV DI,00H
   MOV AX,SI
   MOV DL,02H
   DIV DL
   MOV CL,AL
RE: MOV AL, [BX+SI]
     MOV AH, [BX+DI]      ; XCHG AL,[BX+DI]
     MOV [BX+DI],AL      ; MOV [BX+SI],AL
     MOV [BX+SI],AH
     DEC SI
     INC DI
     DEC CL
     JNZ RE
     MOV AH,09H
     MOV DX, OFFSET STR
     INT 21H
```

8- Write complete ALP which will input the user name from the keyboard. If the user is “**al-ashker**”, it will output “**the username is valid**” else it will output “**Invalid user name**”

*****GOOOOOOOOOOOOOOD LUCK*****